The Efficacy of RANS Models for the Modelling of Turbulence Generating Grids

An Undergraduate Thesis

Presented in Partial Fulfillment of the Requirements for Graduation with Distinction in Aerospace Engineering

By

Austin Michael Hendrickson

The Ohio State University

2021

Defense Committee:

Dr. Randall Mathison, Advisor

Dr. Mike Dunn

Abstract

Turbulence generating grids are a widely used method for intentionally producing various amounts of turbulence in wind tunnels. The analysis of turbulent flow generated by these grids through experimental methods is well documented but has only been computationally investigated primarily through computational fluid dynamics (CFD) using scale-resolving models such as direct numerical simulation (DNS). Because these models are time intensive, the incorporation of lower fidelity Reynolds' Averaged Navier Stokes (RANS) models in CFD can allow for quicker convergence times and greater relaxation of mesh requirements during evaluation of turbulence generating grids. In this work, an unstructured tetrahedral mesh was incorporated with the RANS models Realizable k-ε and SST k-ω to predict the amount of turbulence intensity downstream of the grid and investigate differences in turbulence production at the plate hole walls. This analysis was verified experimentally with the collection of turbulence data retrieved using hotwire anemometry in a blowdown wind tunnel. It was found that the utilization of RANS models presented significant error (>50%) and were not deemed viable for the prediction of downstream turbulence intensity. This study adds to existing literature a more detailed understanding of the flow field of RANS models for analysis of turbulence generating grids, and the results provide a framework of where fast analysis of turbulence generating grids using RANS models may be used in future study and industry application.

Acknowledgments

I would like to acknowledge and thank my advisor Dr. Randall Mathison. Dr. Mathison was always willing to discuss the project and offer incredibly constructive and helpful suggestions. He was adamant on improving not only the project, but myself as a researcher.

I would also like to thank graduate student Spencer Sperling. He never failed to answer all of my relentless questions, take time out of his day to run experiments, or offer sincere guidance for both the project and myself, as a student.

And of course, my endless thanks go to all of my friends and family. Without them, I would not have received the direction or counsel to be the person I am today. All my successes are as much theirs as they are mine.

Table of Contents

List of Tables

List of Figures

Nomenclature

A	Cross-Sectional Area (m^2)
C	Courant Number
d	Pore Diameter (m)
DES	Detached Eddy Simulation
DNS	Direct Numerical Simulation
k	Turbulent Kinetic Energy (W/m^3)
I	Turbulence Intensity (%)
LES	Large Eddy Simulation
m	Mass (kg)
$\dot{m}$	Mass Flow Rate (kg/s)
P	Static Pressure (Pa)
P_o	Tank Total Pressure (Pa)
R	Universal Gas Constant (J/kg·K)
RANS	Reynolds' Averaged Navier Stokes'
SST	Shear Stress Transport
t	Time (s)
T	Temperature (K)
T_o	Total Temperature (K)
u	Mean Flow Velocity Fluctuations (m/s)
U	Mean Flow Velocity (m/s)
V_T	Tank Volume (m^3)
x	Distance Downstream of the Grid (m)
$y+$	Non-Dimensional First Cell Height
γ	Ratio of Specific Heats
ε	Turbulence Dissipation Rate (m^2/s^3)
υ	Kinematic Viscosity (m^2/s)
ρ	Density of Air (kg/m^3)
τ	Shear Stress (Pa)
ω	Specific Turbulence Dissipation Rate (1/s)
Λ_x	Integral Length Scale (m)

Chapter 1. Introduction

1.1 Characteristics of Turbulence

Turbulent flow is characterized by seemingly random flow behavior. This phenomenon is not random however but is an example of a chaotic system. Chaotic behavior in any given dynamic system is specified by large changes in output from very small changes in input. For turbulent flow, these changes are shown in flow velocity and vorticity, with the input being any given geometry or pressure perturbation.

Fully developed turbulence is characterized by constant fluctuation to the mean flow velocity. The amount and magnitude of these fluctuations are quantified as turbulence intensity. Turbulence intensity I in the streamwise direction is defined as:

$$I = \frac{\langle u \rangle}{\overline{U}} \tag{1.1}$$

Where $\langle u \rangle$ is the standard deviation of a set of velocity fluctuations, and $\overline{U}$ is the mean streamwise velocity (Pope, 2000). Turbulence intensity is a useful indicator of the magnitude of turbulence a given flow exhibits.

Another signature characteristic of turbulence is the presence of rotational structures known as eddies. Kinetic energy enters the turbulence at the largest scales of motion, which presents itself as large eddies. The largest eddies then break up and transfer their kinetic energy to smaller eddies. This process is repetitive and is known as the energy cascade. (Pope, 2000). The smallest of these scales, where the eddies are subject to viscous dissipation into heat, are known as the Kolmogorov microscales. (Pope, 2000). The length scale is defined as the length of the eddy, while the ratio of the length to the velocity is the time scale of the eddy. The length scale, Λ_x, can be prescribed as such (Roach, 1987):

$$\Lambda_x = U \int_0^\infty R(T) dT \quad (1.2)$$

Where *U* is streamwise velocity and *R(T)* is an autocorrelation function that considers correlation between velocity fluctuations at different instances of time.

1.2 Turbulence Generating Grids

For over 80 years, screens or grids have been a popular way for intentional generation of turbulence in wind tunnels. Their popularity may be a result of their intrinsic simplicity. When placed normal to the flow in wind tunnels, turbulence grids can generate nearly homogeneous isotropic turbulence downstream of the grid; their position can be manipulated to consistently generate turbulence with a desired intensity and length scale (Vita, 2016). There are two main types of turbulence generators, active and passive. Active generators have a moving boundary in some sense, and inherently work by having the capability of adding momentum to the flow (Gad-el-Hak & Corrsin, 1974).

Passive grids, on the other hand, work to generate turbulence by presenting an obstacle to the flow. This obstacle will result in a pressure drop across the grid (Roach, 1987) and introduce vortex shedding produced by the separation and reattachment of boundary layers created on the grid that impose mean velocity fluctuations on the flow (Loehrke & Nagib, 1972). There are a multitude of passive grid types that have been investigated in the past. They are as follows; as discussed in the literature (Roach, 1987):

(i) square-mesh arrays of round rods or wires (SMR)

(ii) square-mesh arrays of square bars (SMS)

(iii) parallel arrays of round rods or wires (PR)

(iv) parallel arrays of square bars (PS)

(v) perforated plates (PP)

(vi) or various permutations and combinations of the above

This project is focused on work involving perforated plates, type (v). For type (v) grids, the geometric conditions that influence the characteristic turbulence generation properties are grid thickness, pore diameter, and cross-sectional geometry of the pores (Roach, 1987).

Turbulence intensity decays exponentially downstream of the grid regardless of grid type. The exponential rate of streamwise turbulence intensity decay is -5/7 (Roach, 1987). Integral length scale, however, has been shown to grow independent of grid type, and is represented as (Roach, 1987):

$$\frac{\Lambda_x}{d} = I\left(\frac{x}{d}\right)^{1/2} \tag{1.3}$$

Where *d* is the diameter of the pore and *x* is distance downstream of the grid. This relationship implies that the largest eddy size in the flow will increase downstream of the grid by an exponential factor of ½.

1.3 Turbulence Generation and Modelling in CFD

The numerical modelling of turbulence has been an issue scientists and engineers have faced for decades. Due to the inherent randomness of the phenomenon, accurately simulating the quality and behavior of turbulent flow is extremely challenging. In order to numerically solve the Navier-Stokes equations directly, the entire range of spatial and temporal scales, down to the Kolmogorov microscales, must be resolved (Coleman & Sandberg, 2010).

Numerical methods with this capability are called scale-resolving methods. These methods range from complete resolution of all scales, such as direct numerical simulation (DNS), to methods that only resolve the largest scales of turbulence, such as large eddy simulation (LES). Scale resolving methods are extremely computationally expensive, often requiring immense time and processing power to reach a solution (Hart, 2016). The computational cost for DNS to resolve these scales, for example, is dependent on Reynolds' number, with a relationship where a doubling of Reynolds' number for a given flow will increase the computational cost by a factor of 11 (Coleman & Sandberg, 2010). Although scale resolving methods are ideal for addressing research questions requiring the utmost accuracy involving turbulence, they are often not feasible. For a more general-purpose solver that strikes a blend of accuracy and speed, turbulence can be modeled using a Reynolds-Averaged Navier-Stokes (RANS) method.

The modelling of turbulence using a RANS model involves solving for the local mean flow field, only resolving the largest spatial and temporal scales. (Hart, 2016). While objectively not as accurate as DNS or other methods that resolve these scales, RANS models have found great use in industry and academia for being computationally economical. Comparatively, a RANS simulation can be orders of magnitude faster, while also having less stringent meshing requirements (Kulkarni et al., 2011). For this reason, the successful implementation of RANS models for downstream analysis of turbulence qualities would prove to expedite time to insight for industrial and academic applications.

The two RANS models that are most often used for modelling flow with complex geometries are the k-ε model and the k-ω model (Frei, 2017). While each model has a focus on a

formulation for turbulent kinetic energy, *k*, they differ in how they measure turbulence dissipation. The k-ε model represents dissipation with the variable ε, which is defined as turbulence dissipation rate and represents the rate at which turbulence kinetic energy is converted into thermal heat (Frei, 2017). Conversely, the k-ω model represents dissipation with the variable ω, defined as specific turbulence dissipation rate. The ω formulation the k-ω model uses has been shown to be sensitive to inlet free-stream turbulence properties (Frei, 2017)**.** Another way the two models differ is in how they resolve the boundary layer on no-slip walls. The k-ε model utilizes wall functions to estimate the flow field below the fully developed region, while the k-ω model resolves the entirety of the boundary layer. Therefore, the k-ω model allows for more accurate predictions of separation that involve adverse pressure gradients (Frei, 2017), such as turbulence generating grids.

Analysis of turbulence generating grids and screens using CFD has been conducted primarily using scale resolving methods such as LES (Liu et al., 2017), IDDES (Zhu et al., 2019), and DNS (Laizet & Vassilicos, 2010). The feasibility of the grid analyses that utilized scale-resolving methods ranged from needing incredible scalability (up to more than 1000 processors) (Laizet & Vassilicos, 2010), or requiring up to 30 days to yield a result (Liu et al., 2017). Less commonly, turbulence grid simulations that utilized a RANS model analyzed the contrary case of turbulence reduction, rather than production (Kulkarni et al., 2011). Furthermore, review of the literature suggests that there has been very little investigation into turbulence production within turbulence generating grids using RANS models. Additionally, most of the computational work has been done on grids of type (i), type (ii), and type (vi), but no

computational analysis was found for grids of type (v) (Laizet & Vassilicos, 2010; Liu et al., 2017; Zhu et al., 2019).

1.4 Project Objective

The primary purpose of this study is to directly evaluate and compare the accuracy of the Realizable k-ε and SST k-ω RANS models at predicting turbulence intensity downstream of a turbulence generating grid. The accuracy of the model for these parameters will be calculated in comparison with experimental turbulence data from a type (v) grid collected with a hotwire anemometry process in a wind tunnel. If the validity of these RANS models for the analysis of type (v) grids is established, future research that requires numerical analysis of turbulence generating grids would become substantially more computationally feasible.

This work will also investigate the differences in turbulence production at the grid, as well as implications that the differing turbulence intensity values will have on heat transfer over a rigid body downstream of the grid. The presentation of these trends will contribute to the understanding of how RANS models handle the production of turbulence with turbulence generating grids, and the consequences of how these different values of turbulence intensity impact heat transfer.

Chapter 2. Methodology

2.1 Experimental Methodology

One of the most important evaluations of any computational model is accuracy. To measure the accuracy of the computational modelling done in this work, values of turbulence intensity were collected experimentally using a hotwire anemometer placed in a blowdown wind tunnel.

2.1.1 Experimental Setup

The blowdown tunnel used for the experimental work was located in the Small Calibration Facility in the Ohio State Gas Turbine Laboratory (GTL). The tunnel consists of a high-pressure supply tank connected to the test section via a fast-acting valve. On the downstream side of the test section resides a vacuum dump tank that has a variable choke which controls the mass flow rate through the tunnel by imposing a change to the cross-sectional area of the exit into the dump tank. The interior of the test section is rectangular and measures 381mm long, 114mm tall, and 76mm wide. The hotwire was placed in the center of the test section, 280mm downstream of the entrance. A visual representation of the tunnel and a schematic can be seen in Figure 1 and Figure 2, respectively.

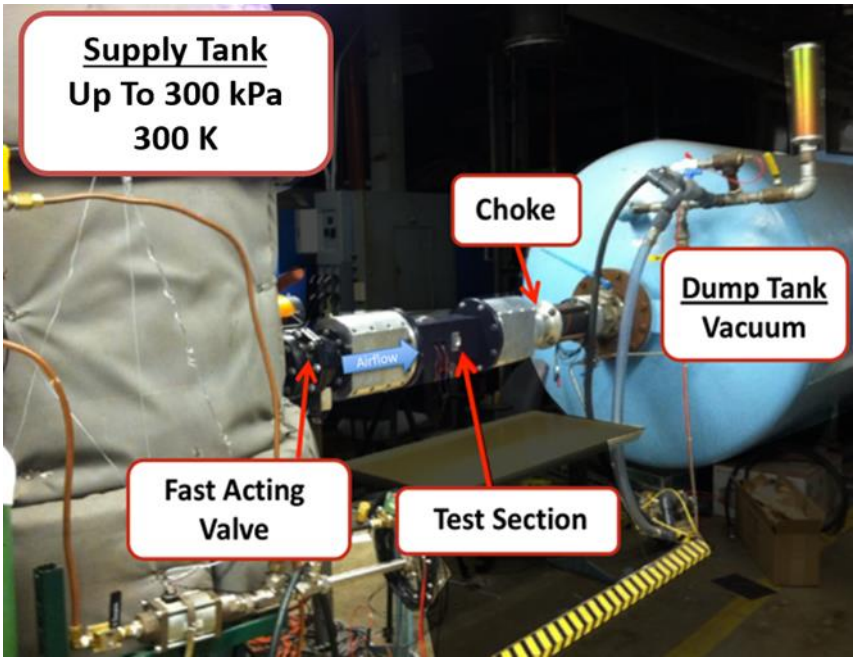

Figure 1. Labelled Photograph of the Small Calibration Facility

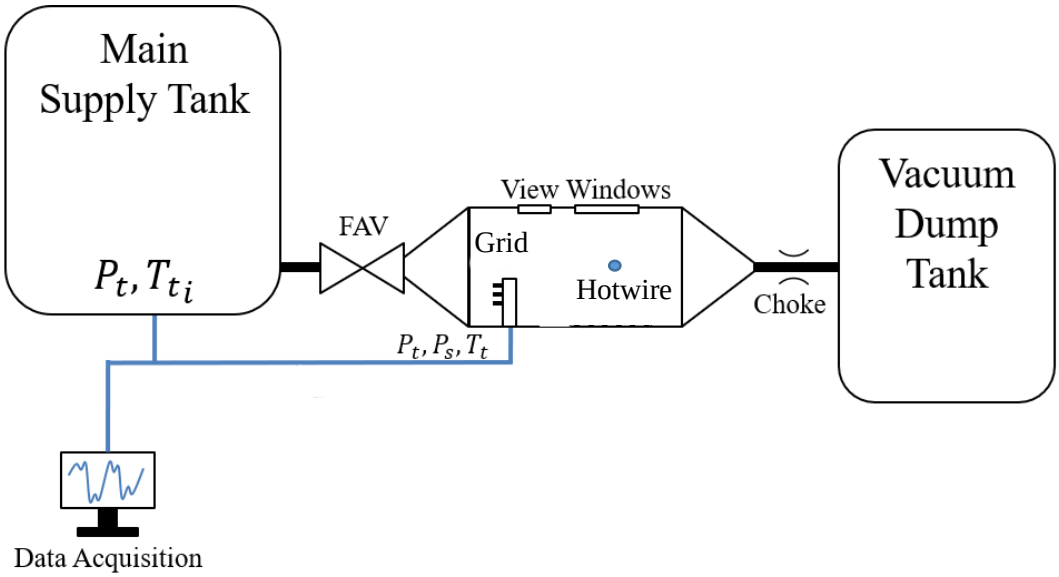

Figure 2. Side-View Schematic of Tunnel

The turbulence generating grid was placed at the beginning of the test section, spanning its width and height. The grid measured 127mm in height, 76.2mm in width, and 6.35mm in depth. The grid used was a perforated plate, type (v), with circular cutouts as the pores in the grid. The pores measured 12.7mm in diameter, spaced 1.3mm vertically apart and 2.4mm apart horizontally. A representation of this grid can be seen in Figure 3.

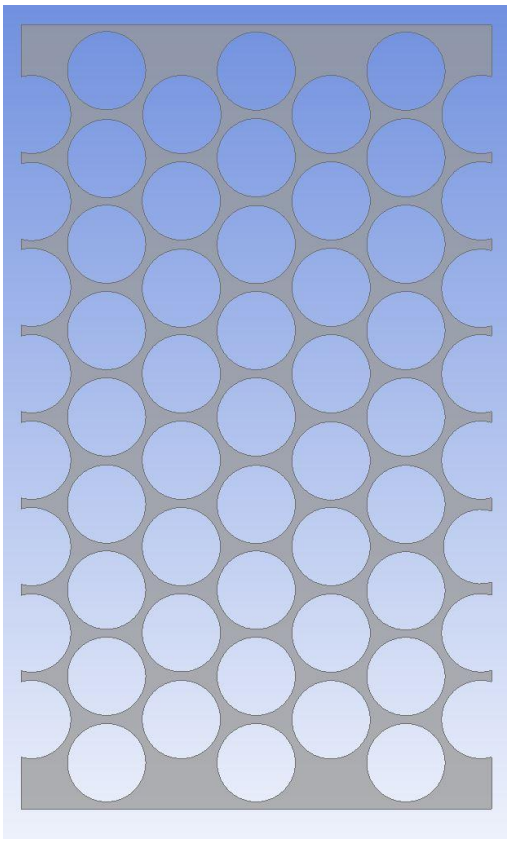

Figure 3. A CAD Drawing of the Grid Used in Experimentation

The hotwire anemometer used was a Dantec 55P14. With a wire probe measuring 5 µm, meant for the measurement of mean flow velocities and flow fluctuations, it is rated for a sampling rate of 100kHz. It was placed into the test section, with its wire oriented perpendicular to the flow direction, via a circular port which seals the hotwire with no chance of releasing air.

A visual representation of the probe can be seen in Figure 4. There were a total of four runs conducted. The first of these runs consisted of a measurement of the no-grid conditions. The subsequent three runs all measured flow properties with the grid in place. All of the grid runs were repeats of one another.

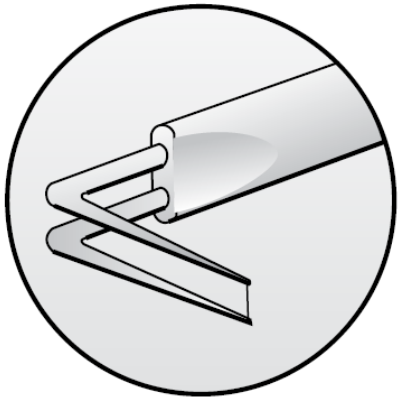

Figure 4. The Dantec 55P14 Hotwire Anemometer

2.1.2 Hotwire Anemometry Reduction

The reduction of voltage data from the hotwire experiments to usable velocity traces, which then could be equated to turbulence data, was done through a MATLAB script. This script can be viewed in Appendix A. Each experimental run consisted of both the voltage hotwire data and already-converted pressure data from the pressure rake inside the test section of the tunnel and the total pressure from the high-pressure tank. The first step in the reduction process was to determine the tank mass over time. This is calculated through equation 2.1

$$m = \frac{{P_o}^{1/\gamma} V_T {P_o}_i^{\frac{\gamma-1}{\gamma}}}{RT_o} \tag{2.1}$$

Where m is tank mass for a given measured tank total pressure P_o, R is the specific gas constant of air, T_o is the total temperature of the air, V_T is tank volume, and γ is the ratio of specific heats for air at sea level. Once the tank mass is derived, a mass flow rate array through the test section was found by imposing a first order *polyfit* function over the data in MATLAB.

Once the mass flow rate array was calculated, a fourth order curve was fit to the mass flow rate data to produce a mass flow rate function with respect to time. To calibrate the hotwire, the voltage data from the hotwire was again fitted with a fourth order curve to the averaged mass flow rate curve. This matched the hotwire data to model the calculated mass flow rate and allow velocity to be derived. Velocity can be derived as in Equation 2.2

$$u_H = \frac{\dot{m}_H}{\rho A} \tag{2.2}$$

Where u_H is hotwire velocity, ρ is the density of air, and A is the cross-sectional area of the test section of the tunnel. The value of $\dot{m}_H$ represents the mass flow array calculated through by the hotwire. Once the hotwire velocity has been calculated, the turbulence intensity may be found through the methods described in Equation 1.1.

2.2 Computational Methodology

Simulations done in this work were carried out in ANSYS Fluent version 19.1 comparing the RANS models Realizable k-ε and SST k-ω. All the simulations were performed on a Linux HPC using a parallel distribution across 8 processing cores. Turbulence intensity and velocity magnitude throughout the entire domain were recorded for each time-step and time to solution for both models was tracked. The aluminum cylinder, to evaluate heat transfer, was added to the model behind the hotwire location. The heat transfer coefficient was then measured across the

cylinder surface and logged with the purpose of comparison between the RANS models. The computational domain can be seen in Figure 5.

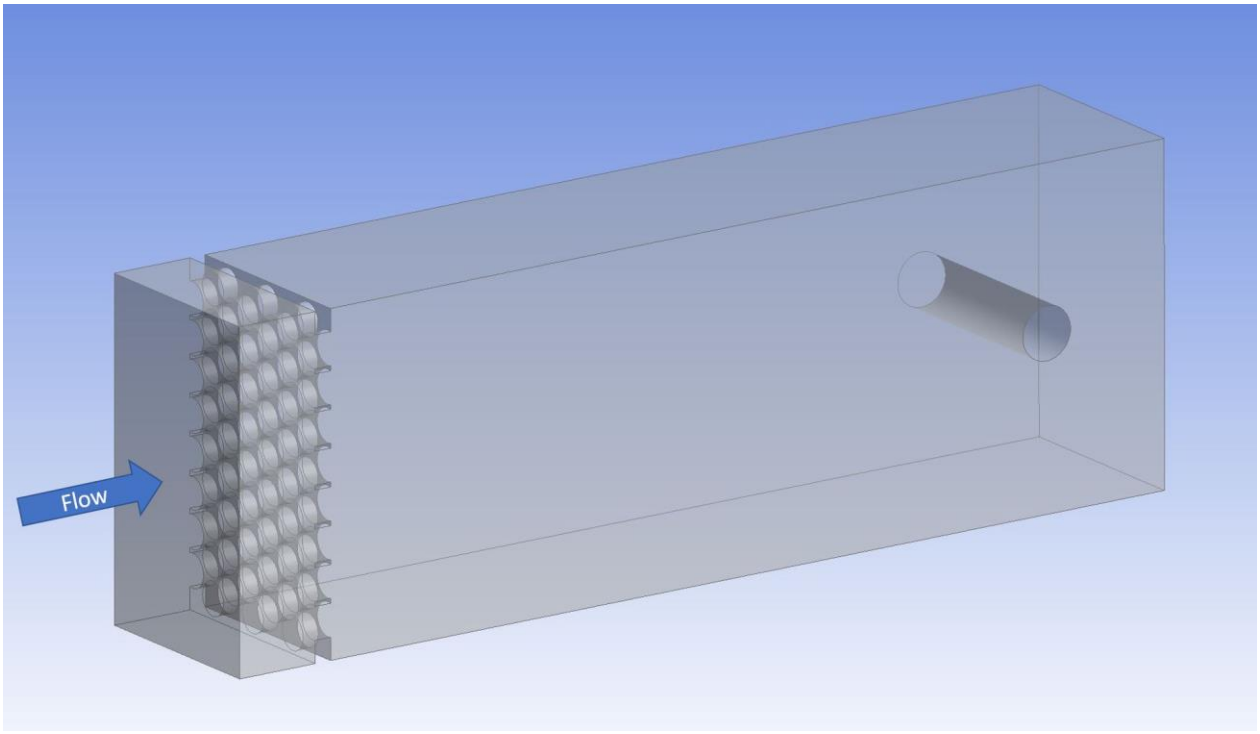

Figure 5. CAD Geometry of the Computational Domain

The computational domain measured the same dimensions as the interior of the test section of the blowdown tunnel used for the hotwire experimentation. The test cylinder measured 20mm diameter, with its center being placed 310mm downstream of the grid. The turbulence grid was created to the specifications mentioned in Section 2.1.1.

2.2.1 Meshing

Different turbulence models in CFD have different requirements for fineness of the mesh needed, with careful consideration of mesh quality needed for wall-bounded flows. For the k-ω model to accurately model the boundary layer for a given flow, it must be able to resolve the laminar sublayer. The height required for the resolution of this sublayer is variable and is

expressed through the non-dimensional quantity $y+$. The formulation for $y+$ can be seen in Equation 2.3.

$$y^+ = \frac{yu_\tau}{\upsilon} \tag{2.3}$$

The value y represents the absolute distance from the wall, while υ is kinematic viscosity. The quantity u_τ is the friction velocity, and is defined as:

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{2.4}$$

With τ_w representing wall shear stress and ρ being density. For the k-ω model to resolve the boundary layer accurately, $y+ \leq 1$. The k-ε model does not resolve the boundary layer; instead, it uses a combination of linear and logarithmic wall functions to represent the boundary layer. The requirement for these wall functions to correctly model the boundary layer is $30 < y+ < 300$. The y+ values for each mesh on the cylinder and grid have been plotted in Figures 6 and 7.

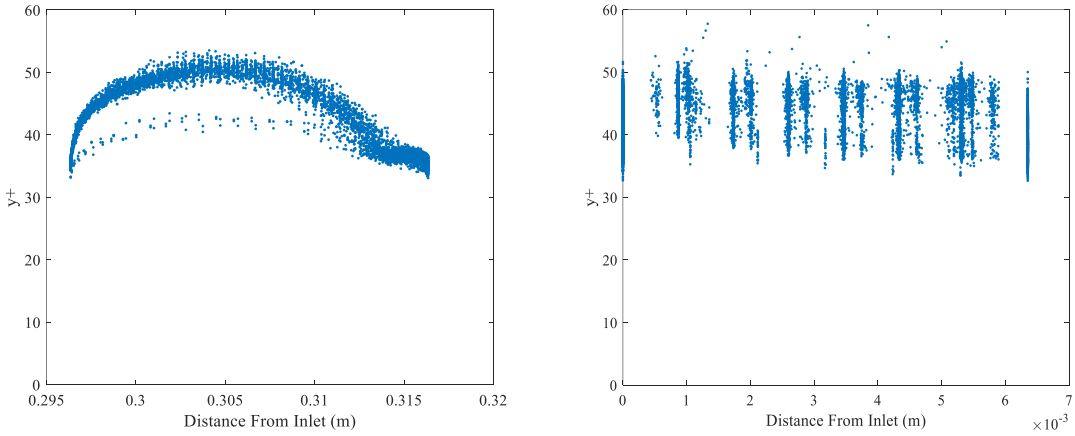

Figure 6. k-ε y+ Value on the Cylinder (left) and Grid (right)

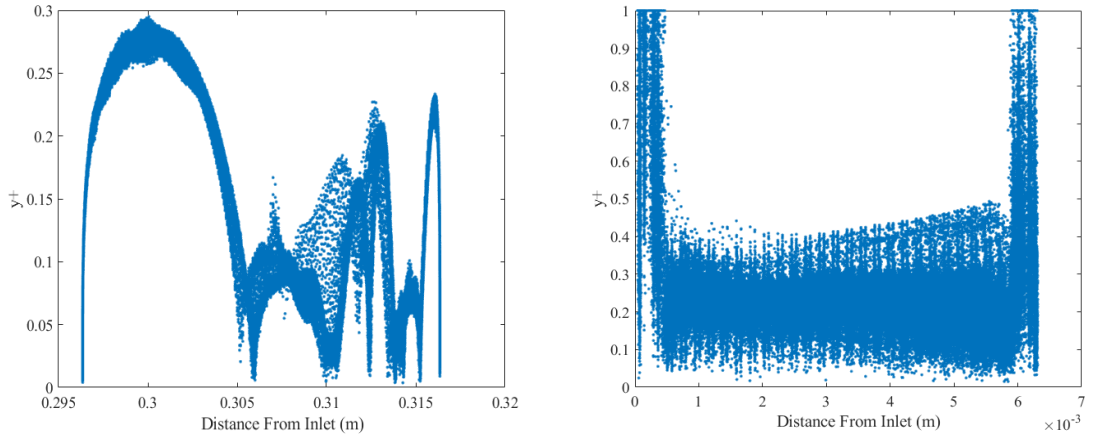

Figure 7. k-ω y+ Value on the Cylinder (left) and Grid (right)

To maximize the specific benefits that each model offers, a different mesh was created for each. Both meshes were unstructured and were composed of tetrahedral elements. The specifications of each mesh can be seen in Table 1 and visuals showing the full mesh and relevant areas of resolution can be seen in Figures 8 through 10.

	Realizable k-ε	**SST k-ω**
Freestream Cell Size	**2.5e-3m**	**2.5e-3m**
Minimum Cell Size	**1e-3m**	**3e-5m**
Cell Count	**1,655,725**	**16,682,571**
Maximum y+ Value	**57.73**	**0.999**
Average y+ Value	**42.81**	**0.414**

Table 1. Mesh Specifications

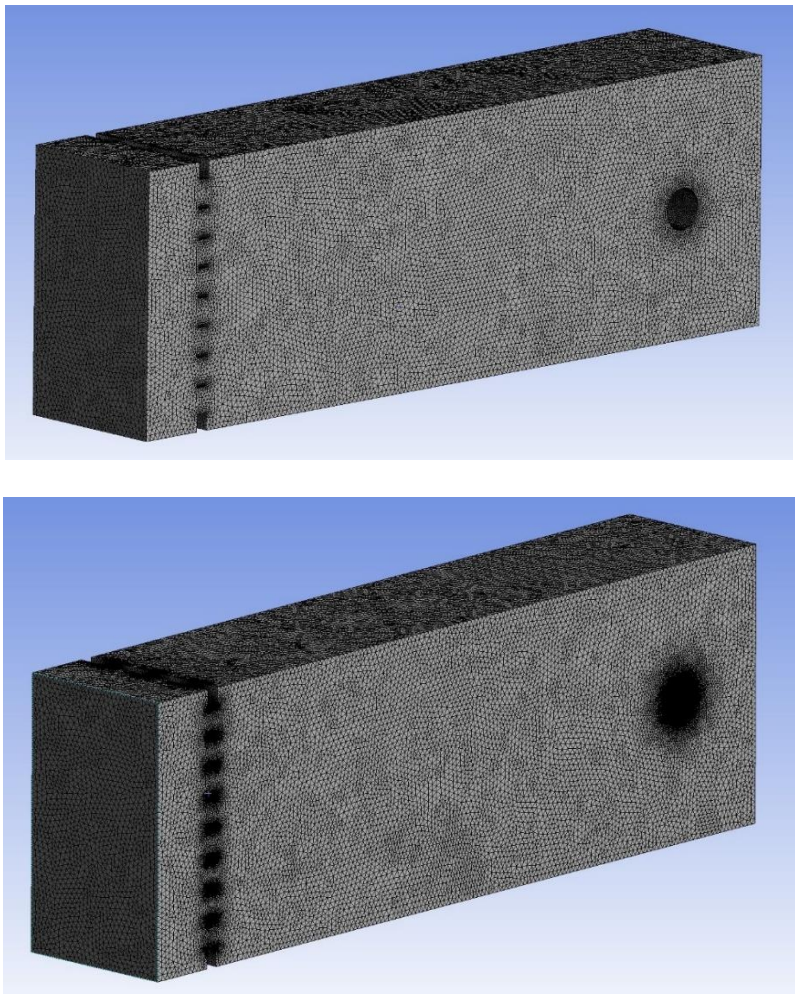

Figure 8. Comparison of Mesh Domains Between k-ε (top) and k-ω (bottom)

In Figure 8, notice the equal sizing of the freestream elements between the two models but higher cell density near the grid and cylinder in the k-ω mesh domain. A closer view of the differences in resolution in these key areas can be seen in the following figures.

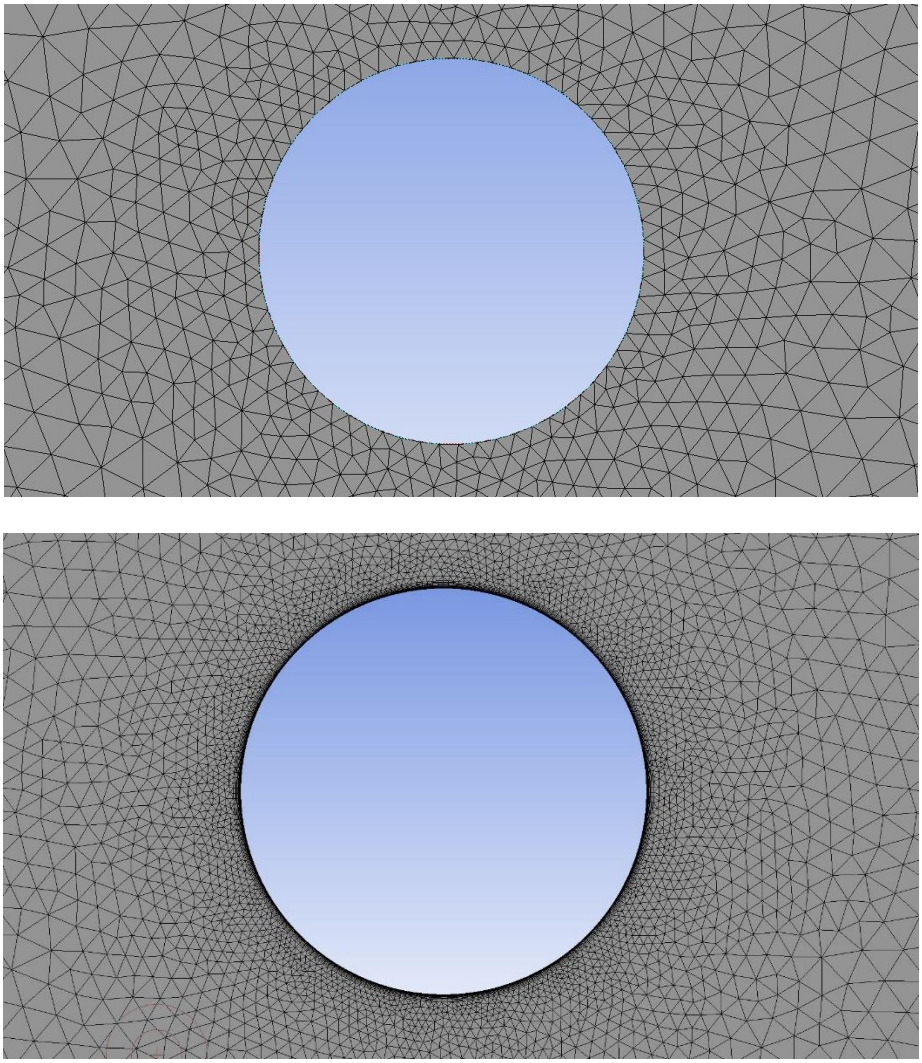

Figure 9. Cylinder Mesh Resolution Difference Between k-ε (top) and k-ω (bottom)

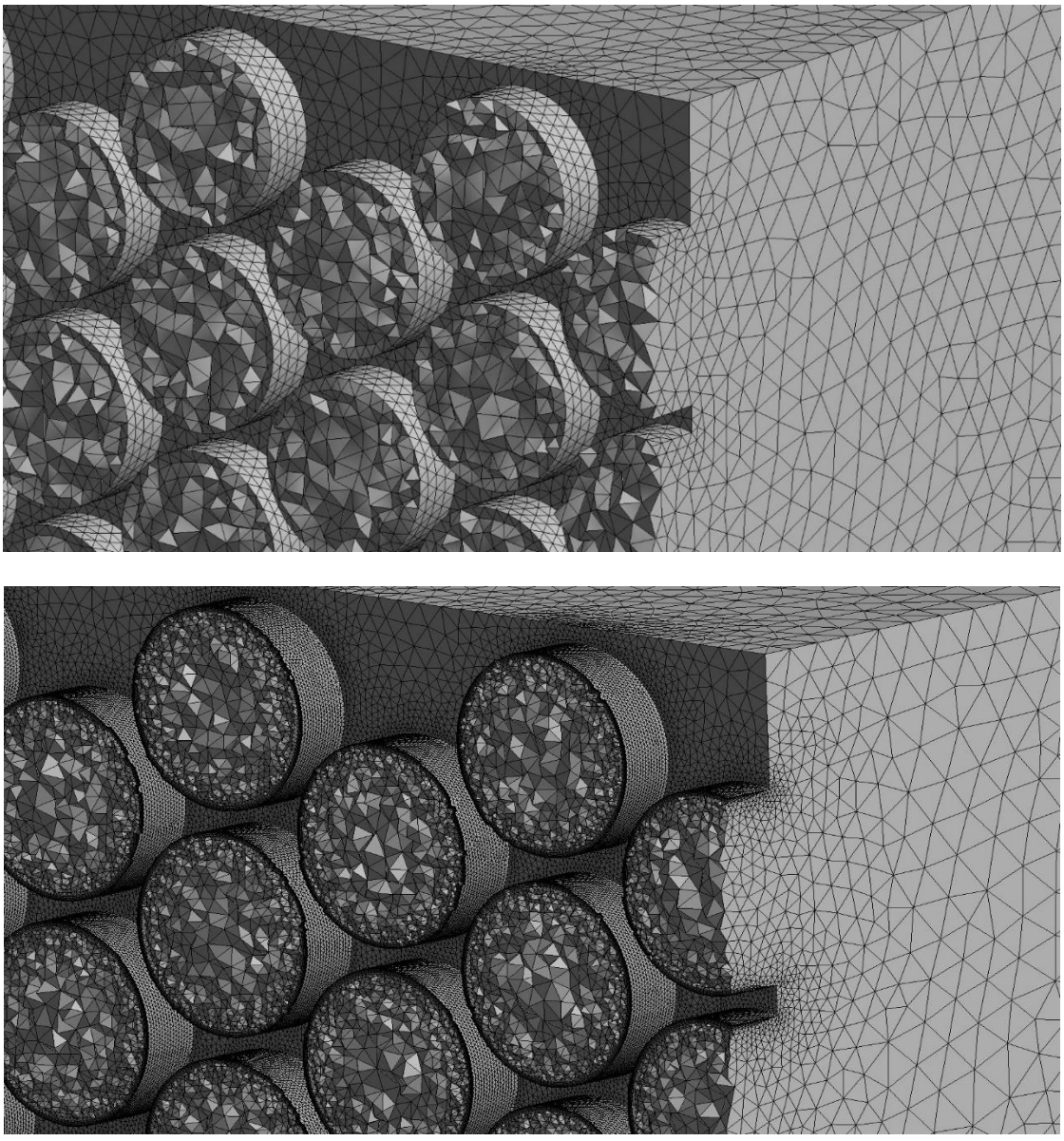

Figure 10. Grid Mesh Resolution Difference Between k-ε (top) and k-ω (bottom)

2.2.2 Setup and Boundary Conditions

The simulation setup consisted of a main velocity inlet and a pressure outlet. The walls of the turbulence grid and the cylinder were set to a no-slip condition. The remaining exterior walls were set to zero-shear as their effect to centerline turbulence intensity is null and imposing a

zero-shear condition allows for a coarse mesh to be implemented, improving convergence time. Inlet boundary conditions of velocity and turbulence intensity were determined by averaging the reduced data from the hotwire experiments discussed in Section 2.1.1. The velocity trace of the experimental run without the grid in place can be seen in Figure 11, where the tunnel was opened around 0.5 seconds after data recording and started to dissipate around 1.5 seconds in.

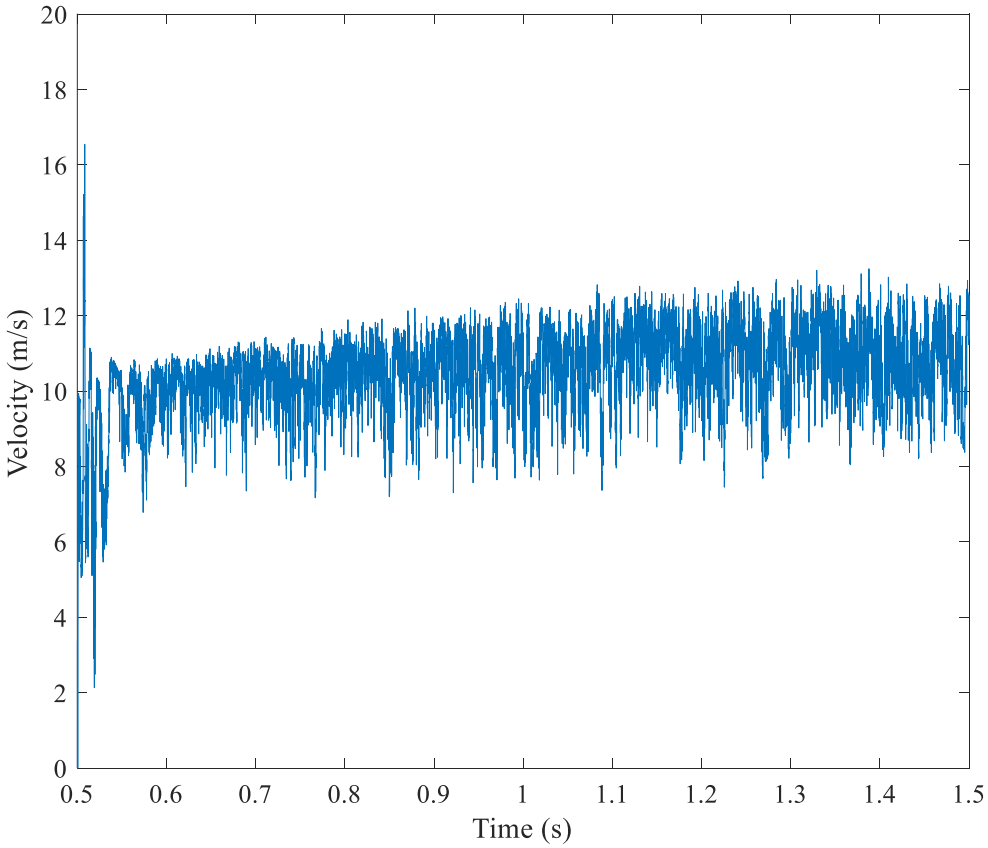

Figure 11. Hotwire Velocity Trace of No-Grid Run

When averaged from 0.6 seconds to 1.5 seconds, to avoid the initial tunnel open outliers, the average velocity was 10.17 m/s. From this value, the inlet velocity in the computational domain

was set to 10.17 m/s. The same has been done for turbulence intensity. The turbulence intensity over time in the no-grid run can be seen in Figure 12.

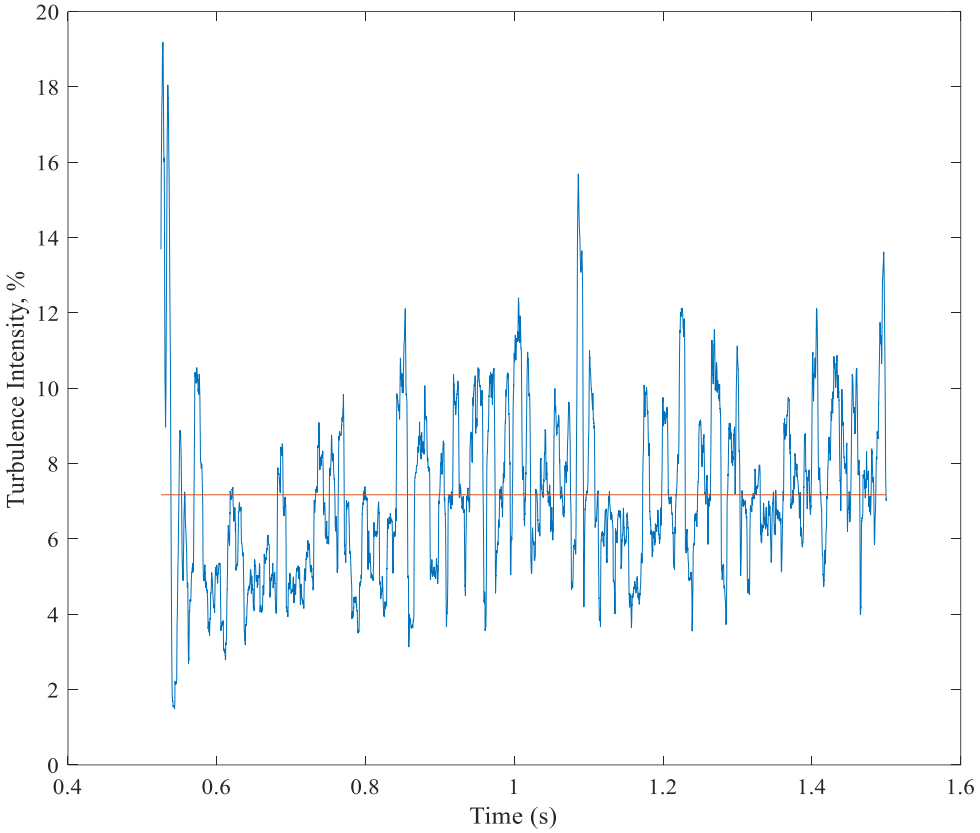

Figure 12. Turbulence Intensity Measured by Hotwire of No-Grid Run

The data was collected from the same time range as the velocity trace was. The average turbulence intensity in this range was calculated to be 7.17%, and the inlet in the computational domain was set as such.

There were two simulations carried out, one using the Realizable k-ε model with enhanced wall functions enabled and the other being the SST k-ω model with low Reynolds' number corrections enabled. The Realizable version of the k-ε model exhibits superior performance with flows that have boundary layers that undergo strong adverse pressure gradients

and separation (Cable, 2009). The SST version of the k-ω model was chosen due to its ability to resolve the boundary layer and transition to the k-ε formulation for the freestream for more accurate prediction of turbulence dissipation (Cable, 2009).

The timestep for each simulation was chosen to minimize time to solution while still exhibiting positive convergence characteristics. The Courant number is the metric that influences convergence with respect to timestep. The Courant number, *C*, is expressed in Equation 2.5.

$$C = \frac{u\Delta t}{\Delta x} \tag{2.5}$$

Where u is the velocity at a given cell, Δt is the timestep, and Δx is a given cell size. To maximize convergence, the Courant number at every cell should be equal to or less than 1. Fundamentally, the Courant number at unity represents a fluid particle travelling the distance of the length of a given cell during the provided timestep. The timestep for the k-ε model was chosen to be 1e-3 seconds, and 1e-4 seconds for the k-ω model. The smaller timestep for the k-ω model was due to the smaller minimum cell size within the k-ω mesh domain. Each simulation ran for a total time of 0.5 seconds. This required 500 timesteps for the k-ε model, and 5000 timesteps for the k-ω model.

To collect results, a point was created for both simulations in the center of the domain, 280mm downstream of the inlet. This point represents the hotwire collection point. Velocity and turbulence intensity data was collected for each time step and exported at the end of the simulation. To measure the implications of heat transfer characteristics that each model exhibited downstream of the grid, total heat transfer rate per timestep from the cylinder was measured for each model. The flow from the velocity inlet was given a temperature of 400K, while the temperature of the cylinder wall was an imposed constant at 300K. These data sets were then

imported, analyzed, and visualized using MATLAB. The entire solution was also exported into CFD-Post to create contour visuals.

Chapter 3. Results

3.1 Experimental Results

The reduced hotwire velocity traces from each run can be seen in Figure 13.

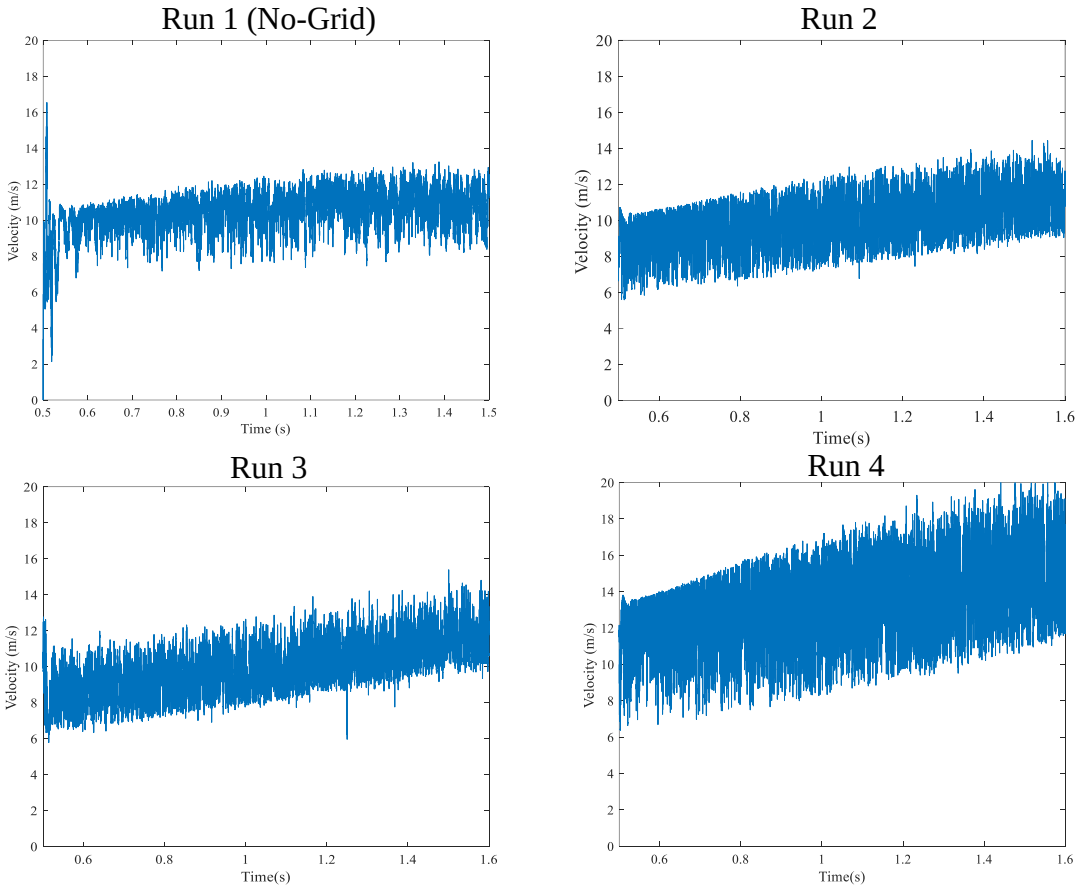

Figure 13. Hotwire Velocity Traces with and without the Grid

The data from each run was collected when the flow through the test section was fully developed. Therefore, the time range for each data set was 0.5s to 1.6s. The turbulence intensity plots, which are derived from these velocity traces, are shown in Figure 14. The average turbulence intensity was calculated and shown in red.

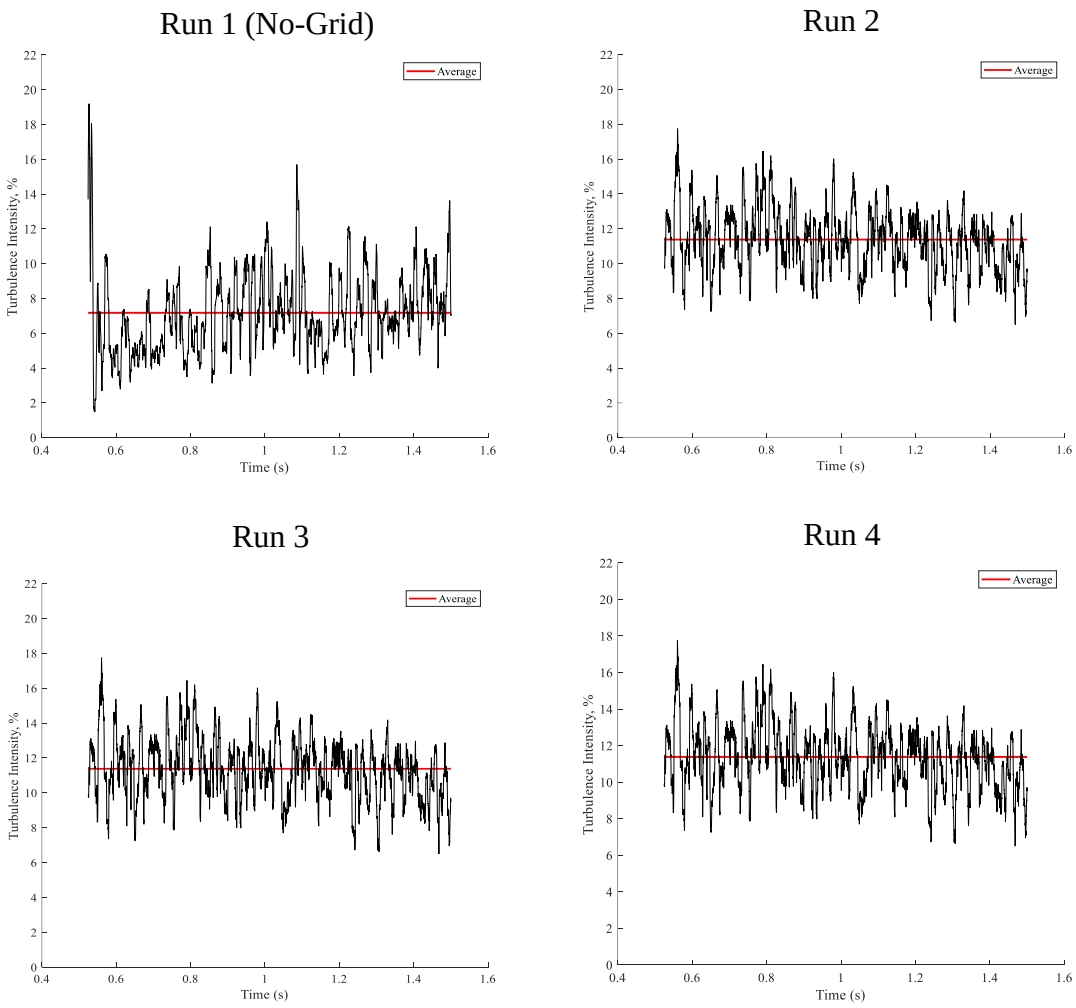

Figure 14. Turbulence Intensity Results from Hotwire

The average turbulence intensity of the first, grid-less run was calculated to be 7.17%. The three grid runs then averaged 11.38%, 10.25%, and 13.57% respectively. The final average turbulence intensity value that will be compared to the computational result will be the average of the concatenation of all grid runs, which results in a total average grid turbulence intensity of 11.73%. These results are tabulated in Table 2.

Grid Runs	Turbulence Intensity
1 (No-Grid)	**7.17%**
2	**11.38%**
3	**10.25%**
4	**13.57%**
Total Grid Average	***11.78%***

Table 2. Experimental Turbulence Intensity Values

3.2 Computational Results

Turbulence intensity contours down the center of the domain have been plotted for each model at t = 0.5s and are shown in Figure 15.

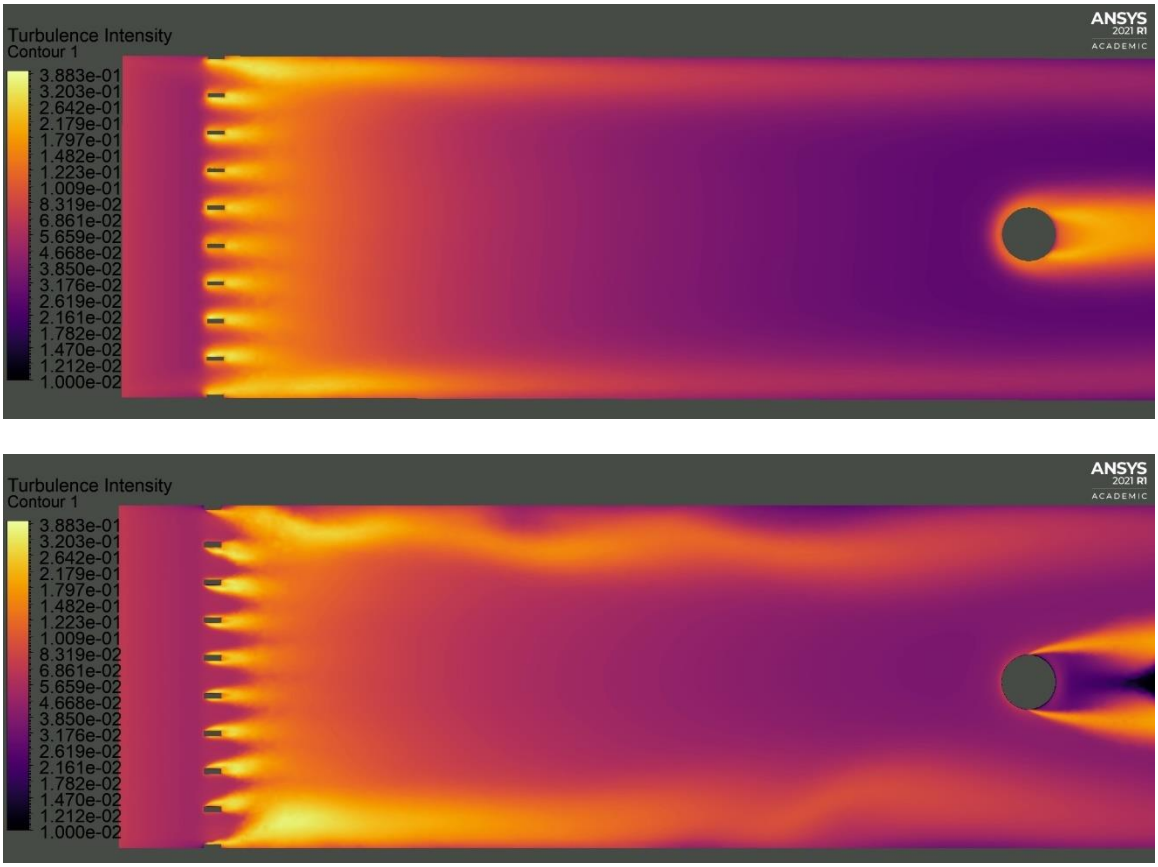

Figure 15. Turbulence Intensity Contours of k-ε (top) and k-ω (bottom) @ t = 0.5s

The maximum turbulence intensity value for the k-ε model was 37.2%, while the k-ω model registered a maximum value of 38.8%. Downstream, at the representative hotwire point surface, turbulence intensity was measured to be 4.14% for the k-ε model and 5.52% for the k-ω model. These values, compared to the experimental average of 11.78%, result in a percent error of 64.7% and 52.9% respectively.

To better visualize the differences in turbulence production between the two models, a contour of *k* production which demonstrates production of turbulent kinetic energy, is visualized in Figure 16.

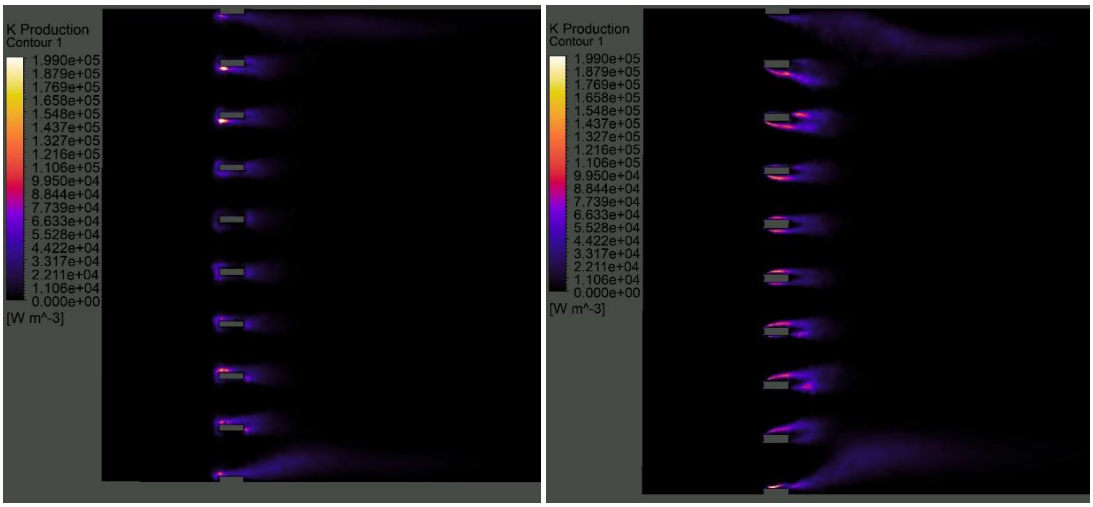

Figure 16. k Production Contour of k- ε (left) and k-ω (right) @ t = 0.5s

The k-ω model showed more overall *k* production when compared to the k-ε model. These differences are the most apparent near the center of the grid.

Velocity contours serve as an intuitive way to visualize the effects that different values of turbulence intensities can have on the flow field, as well as the differences in separation tendencies that each model possesses. The velocity contours for each model can be seen in figure 17.

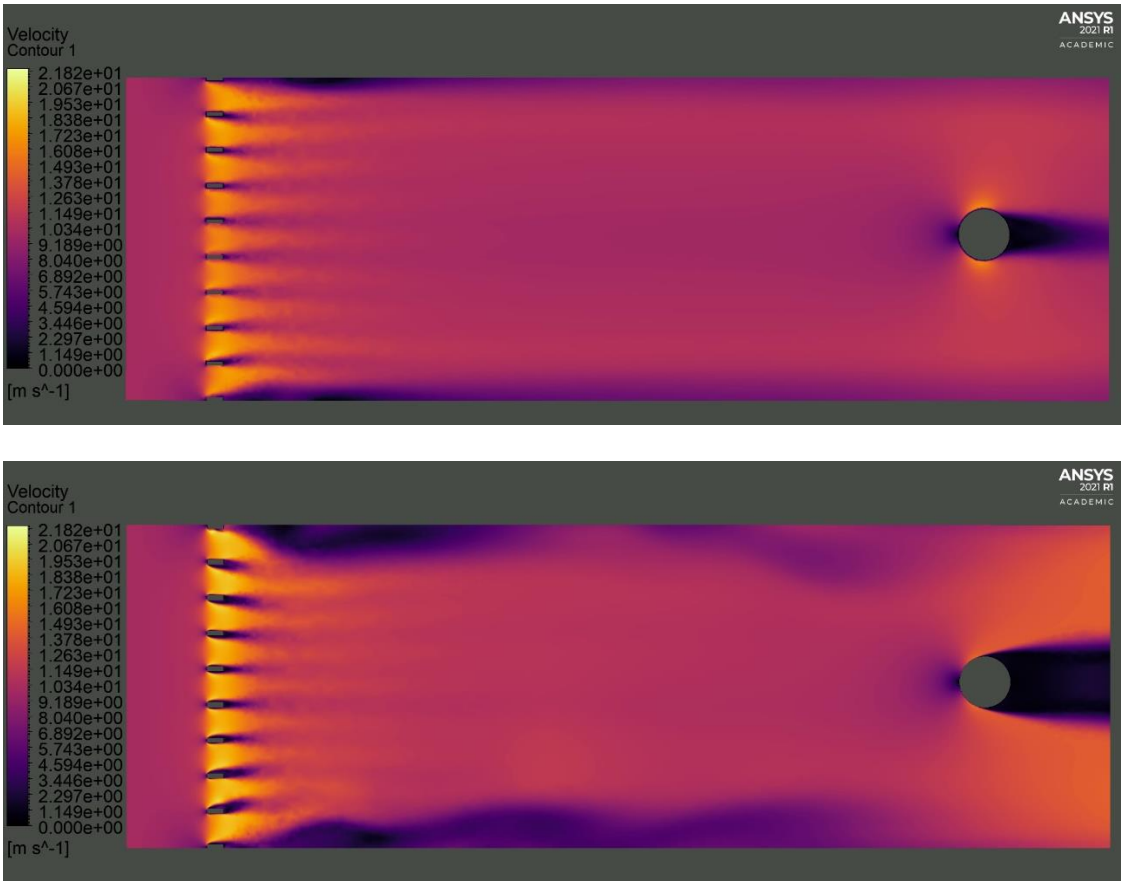

Figure 17. Velocity Contour of k-ε (top) and k-ω (bottom) @ t = 0.5s

The k-ω possessed both more aggressive separation characteristics and higher flow acceleration through the pores of the turbulence grid, resulting in a maximum velocity of 21.52m/s. The k-ε model predicted a maximum velocity of 19.11m/s.

Wall heat flux contours were produced to visualize the consequences that the differences each model had on heat transfer. The measured heat flux leaving the cylinder, viewed from the top of the domain looking down, can be seen in Figure 18.

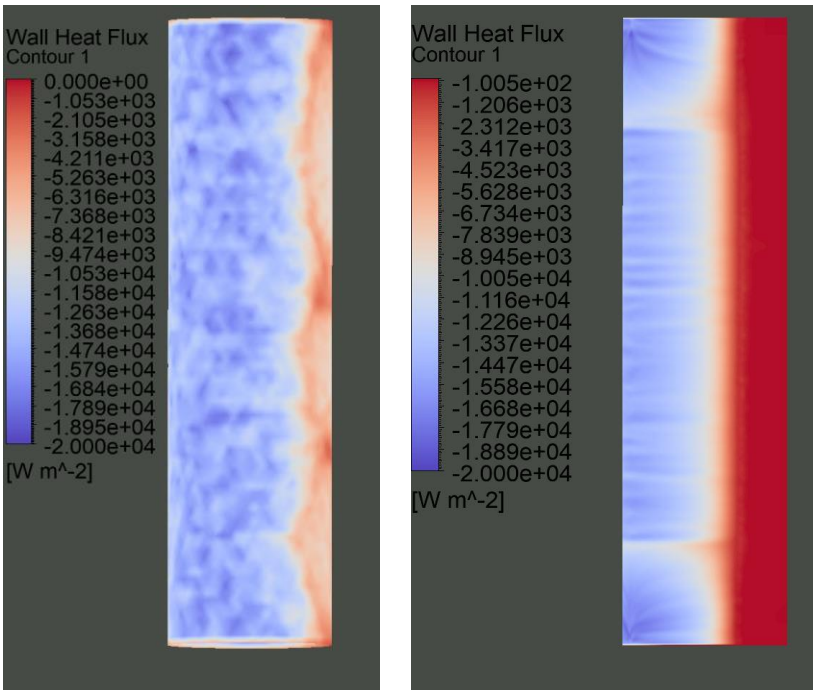

Figure 18. Measured Heat Flux Contour of k-ε (left) and k-ω (right) @ t = 0.5s

The maximum heat flux out of the cylinder was nearly identical between the two models, registering at 19,873 W/m^2 for the k-ε model and 19,446 W/m^2 for k-ω. Both models also predicted less heat transfer in regions of separated flow. The final results, prescribed in total heat transfer rate into the cylinder, was measured as 54.3W for the k-ε case, and 26.5W in the k-ω case. The total simulation time was 202 minutes for k-ε and 1110 minutes for k-ω. The results presented in this chapter, as well as simulation run time, have been combined and presented in Table 3.

	Turbulence Intensity (TI)	**TI Percent Error from Hotwire**	**Total Heat Transfer Rate**	**Total Simulation Run Time**
Realizable k- ε	**4.14%**	**64.7%**	**54.3W**	**202 Minutes**
SST k- ω	**5.52%**	**52.9%**	**26.5W**	**1110 Minutes**

Table 3. Final Computational Results

Chapter 4. Discussion

The poor accuracy of the computational values gathered at the representative hotwire point indicates that these two RANS models are not viable methods for the prediction of downstream turbulence intensity from a turbulence generating grid. The consideration of this information and various turbulence values over the entire domain allows inference on why these models failed and where they may succeed.

When viewing the turbulence intensity values in Figure 15, it is evident that both models produced high values of turbulence intensity near the grid, with a maximum between 37% and 39% for both models. However, the downstream intensity values of 4.14% for k-ε and 5.52% for k-ω are much lower than the experimental average of 11.72%. It can be insinuated that the low values of intensity downstream are a result of excessive turbulence dissipation due to the time averaging qualities that RANS models possess. This also implies that the magnitude of the intensity values reported by the experimental hotwire runs were a result of the smaller scales of turbulence presenting local velocity fluctuations at the hotwire position. It is possible that both models would be more accurate when considering near-grid turbulence values; however, this conclusion cannot be specifically drawn from the scope of this work.

The higher value of turbulence intensity reported from the k-ω model compared to the k-ε model can be attributed to the higher predicted turbulence production at the grid. While difficult to observe in the turbulence intensity contour in Figure 15, a better visualization of this result can be seen in the *k* production contour in Figure 16. This figure shows higher overall production of turbulent kinetic energy from each pore in the grid by the k-ω model when compared to the k-ε model. The explanation of higher turbulence production from the k-ω model lies in its higher

accuracy in the resolution of complex boundary layer interactions, meaning it has a more aggressive prediction of separation in the face of adverse pressure gradients which directly leads to more turbulence generation. This separation behavior, while significant at the turbulence generating grid, was also exhibited on the surface of the cylinder which can be most easily visualized in Figure 17 showing a more forward point of separation in the k-ω model. Since the SST k-ω model switches to a k-ε formulation in the freestream, the freestream dissipation rate is identical between both models. This directly points to the higher turbulence production at the grid being responsible for the higher value of downstream turbulence intensity that the k-ω model presented.

The effects of differing turbulence intensity between models on heat transfer are inconclusive. Considering the difference of turbulence intensity at the collection point was 1.38% while the difference in total heat transfer from the cylinder varied by a factor of 2, it is more appropriate to attribute the change in heat transfer to the differences in point of separation between the two models. The k-ω model showed a more forward point of separation on the cylinder body, evident in Figure 17. The consequences of this larger stagnation region can be seen in Figure 18. Due to the recirculation of air in the stagnation region, less area of the cylinder was consistently exposed to the hotter free stream flow, and therefore less total heat transfer was observed. Alternatively, the k-ε model had a further-back point of separation and this resulted in double the total heat transfer. Therefore, it can be concluded that the k-ω model is more desirable for heat transfer on rigid bodies that present adverse pressure gradients to a flow.

The time to solution for the k-ω simulation was 5.49x longer than the k-ε run. This increase in time was lower than expected, as the k-ω mesh contained 10.07x the number of

elements with a timestep that was 10x shorter, and therefore required 10x the amount of timesteps to reach an equal absolute time. This scaling comes from the fact that the k-ω model took less iterations per time step to converge versus the k-ε model. While not explicitly tracked in this work, it appeared as though the k-ε simulation took roughly 25 iterations per timestep to complete, while the k-ω run took only 3-4. However, this still suggests less-than-linear scaling with element count between the two simulations. Regardless, the longer k-ω simulation was up to 10.8x faster than a similar setup using a scale resolving method, which used 10x the processing cores (Liu et al., 2017). Ultimately, the time saved by using these RANS models does not justify their use for downstream prediction of turbulence intensity from a turbulence generating grid because no reliable conclusion can be drawn from the magnitude of their percent error values.

4.1 Limitations and Future Work

The results of this work were limited to the study of downstream turbulence predictive accuracy of the RANS models of k-ε and k-ω. The inability to quantify the accuracy of the RANS models for near-grid turbulence intensity is due to the single location placement of the hotwire runs in the tunnel. Further work involving multiple hotwire placements downstream of the tunnel would be valuable to evaluate the accuracy of the prescribed RANS models at different points downstream of the grid. Additionally, work involving the investigation of different variants of the k-ε and k-ω models is also recommended. Hybrid RANS models, such as the SST SAS k-ω model, resolve scales of turbulence where the mesh resolution is fine enough

and revert to a RANS formulation elsewhere. This type of model may strike a perfect balance of speed and accuracy for the types of systems in this work.

Another notable limitation is that no experimental heat transfer data was collected. The inclusion of a cylinder in the test section of the tunnel, with other appropriate boundary conditions set, would allow the computational work done in this study to be evaluated for physical accuracy. On top of this, no mesh independence study was conducted for the computational mesh as well, meaning it is possible that simulation times for both models could have been improved while producing the same results.

Chapter 5. Conclusion

The Realizable k-ε and SST k-ω RANS models have been tested for prediction of downstream turbulence intensity produced by a turbulence generating grid. These models, as measured, were found to not be viable methods for assessing downstream turbulence characteristics. While the studied RANS approach did not achieve efficacy for the prescribed results, other implications may be gathered from the data collected. It can be implied that the magnitude of the hotwire turbulence intensity came from the smaller scales of turbulence that the RANS models did not resolve.

Though the effects of turbulence intensity on heat transfer were inconclusive, the importance of separation on a rigid body in fluid flow was reinforced by the contrast between the two models. While the two specific models used in this work were proven to not be viable for downstream analysis of turbulence intensity from a grid, this does not disprove the efficacy of RANS models for all potential analysis of similar systems. RANS models may still find use in near grid analysis and analysis into the turbulence production on the grid itself.

References

Cable, M. (2009). *An Evaluation of Turbulence Models for the Numerical Study of Forced and Natural Convective Flow in Atria*. *May*, 1–136.

Coleman, G. N., & Sandberg, R. D. (2010). A primer on direct numerical simulation of turbulence - methods, procedures and guidelines. *Technical Report*, 1–21. http://eprints.soton.ac.uk/66182/

Frei, W. (2017). *Which Turbulence Model Should I Choose for My CFD Application?* https://www.comsol.com/blogs/which-turbulence-model-should-choose-cfd-application/

Gad-el-Hak, M., & Corrsin, S. (1974). Measurements of the nearly isotropic turbulence behind a uniform jet grid. *Journal of Fluid Mechanics*, *62*(1), 115–143. https://doi.org/10.1017/S0022112074000607

Hart, J. (2016). Comparison of Turbulence Modeling Approaches to the Simulation of a Dimpled Sphere. *Procedia Engineering*, *147*(0), 68–73. https://doi.org/10.1016/j.proeng.2016.06.191

Kulkarni, V., Sahoo, N., & Chavan, S. D. (2011). Simulation of honeycomb-screen combinations for turbulence management in a subsonic wind tunnel. *Journal of Wind Engineering and Industrial Aerodynamics*, *99*(1), 37–45. https://doi.org/10.1016/j.jweia.2010.10.006

Laizet, S., & Vassilicos, J. C. (2010). Direct numerical simulation of fractal-generated turbulence. *ERCOFTAC Series*, *13*, 19–25. https://doi.org/10.1007/978-90-481-3652-0_3

Liu, L., Zhang, L., Wu, B., & Chen, B. (2017). Numerical and experimental studies on grid-generated turbulence in wind tunnel. *Journal of Engineering Science and Technology Review*, *10*(3), 159–169. https://doi.org/10.25103/jestr.103.21

Loehrke, R. I., & Nagib, H. M. (1972). *Experiments on Management of Free-Stream Turbulence*

DISTRIBUTION AND AVAILABILITY ON BACK COVEiK. *598*.

Pope, S. (2000). Turbulent Flows. In *European Journal of Mechanics - B/Fluids* (Vol. 21, Issue 1). https://doi.org/10.1016/s0997-7546(01)01166-9

Roach, P. E. (1987). The generation of nearly isotropic turbulence by means of grids. *International Journal of Heat and Fluid Flow*, *8*(2), 82–92. https://doi.org/10.1016/0142-727X(87)90001-4

Vita, G. (2016). *Generating turbulence in Wind tunnel experiments: use of passive grids 1*. *February 2016*, 1–14. http://www.winercost.com/cost_files/STSM_Report-Vita.pdf

Zhu, W., Xiao, Z., & Fu, S. (2019). *Simulations of turbulence screens for flow and noise control in tandem cylinders*. https://doi.org/10.13009/EUCASS2019-289

Appendix A. MATLAB Code

```
% Hotwire Reduction Code

load('kulitecals0904.mat')
load('run4.mat');

hwv=dataRaw(3:end,3);
mpv=dataRaw((3:end),4);
psv=dataRaw((3:end),5);
ptv=dataRaw((3:end),6);
ttv=dataRaw((3:end),7);

[n,~]=size(hwv);
mp=zeros(n,1);
ps=zeros(n,1);
pt=zeros(n,1);
tt=zeros(n,1);
ts=zeros(n,1);
Mach=zeros(n,1);
mass=zeros(n,1);
mdot=zeros(n,1);
u_Mach=zeros(n,1);
u_Mass=zeros(n,1);

Vmain=0.72; %m^3 -- liang's paper
gamair=1.4;
rair=0.287;
T0air=300;

% toff=0.003216;
toff=0;
mp = mpv;
pt = ptv;
ps = psv;
ts = ttv;

for i=1:n

    mass(i)=mp(i)^(1/gamair)*Vmain*mp(1)^((gamair-1)/gamair)/rair/T0air;


    if mp(i)/ps(i) >0
        Mach(i)=((mp(i)/ps(i))^(.4/1.4)-1)*2/.4;
    end

    if Mach(i)>0
        Mach(i)=sqrt(Mach(i));
    else
        Mach(i)=0;
    end
```

```
    u_Mach(i)=Mach(i)*sqrt(1.4*287*ts(i));


end

mass=smooth(mass,250);

time=linspace(0,10,n)';

%%

figure,plot(time,ps,time,pt,time,mp), title('Rake
Pressures'),legend('ps','pt','mp')
figure,plot(time,Mach),title('test section mach number')
figure,plot(time,mass),ylabel('tank mass'),xlabel('time'), title('Tank Mass
vs. Time');
figure,
plot(time,ps,time,mp),ylabel('pressure,kPa'),xlabel('time'),legend('ps','mp')
, title('static, tank pressure vs time');

%%
range=500;
for i=range/2+1:n-range/2

    P=polyfit(time(i-range/2:i+range/2),mass(i-range/2:i+range/2),1);
    mdot(i)=-1*P(1);

    u_Mass(i)=mdot(i)/(1000*ps(i)/287/ts(i)*.007319);

end
%%

figure,plot(time,mdot), title('mass flow rate vs. time');
figure,plot(time,u_Mass,time,u_Mach), legend('mass vel', 'mach vel'),ylim([-
20,200]), title('velocity from mass and mach vs time');

%%
[n,~]=size(hwv);
time=linspace(0,3,n)';
n1=70000;
n2=300000;
n=n2-n1+1;

x=linspace(.7,1.1,n);

mdoteqn=polyfit(time(n1:n2),mdot(n1:n2),4);
mdotfit=polyval(mdoteqn,time(n1:n2));
figure,plot(time,mdot,time(n1:n2),mdotfit),title('mass flow')
```

```
vcal=polyfit(hwv(n1:n2),mdotfit,4);
mdotcalc=polyval(vcal,x);
mdothw=polyval(vcal,hwv);

figure,plot(hwv(n1:n2),mdotfit,x,mdotcalc), title('Mass Flow Calibration Run
3')
figure,plot(time,mdothw,time,mdot), title('Mass Flow From Hotwire and Tank
Pressure'), ylim([-1, 1]), xlabel('Time'), ylabel('Mass Flow (kg/s)'),
legend('Hotwire','Main Pressure');
%%
[n,~]=size(hwv);
uhw=zeros(n,1);
for i=1:n
    uhw(i)=mdothw(i)/(1000*ps(i)/287/ts(i)*.007319);
end

figure,plot(time,u_Mass,time,uhw), legend('mass', 'hotwire'), ylim([-50,
50]), title('Velocity from Hotwire and Mass Flow');

%% Turbulence Intensity Code

mdhw=polyval(vcal,hwv);

[n,~]=size(hwv);

range = 1000;

tu=zeros(n,1);

for i=range/2+1:n-range/2

    tu(i)=100*std(mdhw(i-range/2:i+range/2))/mean(mdhw(i-range/2:i+range/2));

end

time=linspace(0,3,n)';

n1=70000;
n2=200000;

AV=ones(n,1)*mean(tu(n1:n2));

figure,
% subplot(2,1,1)
hold on
plot(time(n1:n2),AV(n1:n2),'r', 'LineWidth', 2);
plot(time(n1:n2),tu(n1:n2), 'k'), ylim([0,22]);
ylabel('Turbulence Intensity, %'), xlabel('Time (s)')
legend('Average');
hold off
```